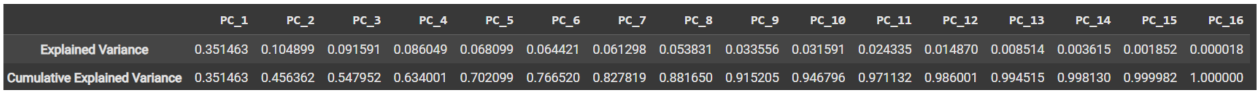
Task 1 Machine Learning Algorithms’ Guidelines

1. Principal Component Analysis (PCA)

1.1 Advantage: PCA helps to simplify the complex datasets with many features by reducing them into a smaller set of important features. It helps to reduce the noise, improves the performance of the predictive models, address multicollinearity problems, and combine information from the original data to generate meaningful features that capture patterns and relationships.

1.2 Basics: The principal component analysis is a dimensionality reduction technique that is used in unsupervised machine learning. It used the covariance of the data features to transform a dataset with many features into a smaller set of new variables that still retain the essential information from the original features.

1.3 Computation: To demonstrate the PCA implementation in Google Collab, we use the data of Corporate credit rating with financial ratios available at Kaggle. We slightly modified the dataset by removing all non-numerical independent variables. There were 16 numerical independent variables remaining in the dataset. Then, we normalized the independent variables by using the standard scaler and then performed the PCA. The results of PCA implementation including the explained variance ratio (EV) for each component and cumulative explained variance ratio (CEV) are demonstrated as the figure below.



	PC_1	PC_2	PC_3	PC_4	PC_5	PC_6	PC_7	PC_8	PC_9	PC_10	PC_11	PC_12	PC_13	PC_14	PC_15	PC_16
Explained Variance	0.351463	0.104899	0.091591	0.086049	0.068099	0.064421	0.061298	0.053831	0.033556	0.031591	0.024335	0.014870	0.008514	0.003615	0.001852	0.000018
Cumulative Explained Variance	0.351463	0.456362	0.547952	0.634001	0.702099	0.766520	0.827819	0.881650	0.915205	0.946796	0.971132	0.986001	0.994515	0.998130	0.999982	1.000000

With the acceptance level of the threshold of 0.9, we could select 9 principal components that have CEV equal to 91.52%. These components will be used to transform 18 original features to 9 new uncorrelated features. The complete results of PCA and orthogonal transformation are presented in the coding file.

1.4 Disadvantages:

- PCA is sensitive to the scale of data. If the data is not standardized, some features that have large scales can dominate the principal components. Moreover, outliers could affect the covariance matrix calculation and distort the principal components.

- There may be some loss of information in the process especially for the low-variance components.

- The principal components are linear combinations of the original features that make them harder to interpret compared to the original features. Additionally, if the features’ relationships are non-linear, the PCA might not be able to sufficiently capture the essential information of the original data.

- PCA is limited to numerical data due to it relies on the computation of the covariance matrix and linear algebra operations. If categorical variables are presented in the dataset, we need to convert them to be numerical data. However, the interpretation of the principal components derived from these categorical variables may not always be meaningful.

- For the large dataset, computation of the covariance matrix, eigenvalues and eigenvectors could take a long time and high computational costs.

1.5 Equations: The core equation in the principal component analysis is related to finding eigenvalues and eigenvectors. This equation plays a crucial role in identifying the principal components that capture the most variance in data.

$$Cv \quad = \quad \lambda v$$

where C is a variance-covariance matrix of the dataset

λ is the eigenvalues (represented variance explained by the principal components)

v is the eigenvectors (represented a principal component)

We could obtain the eigenvalues by solving the characteristic equation of $|C-\lambda I| = 0$. Then, the obtained eigenvalues will be utilized in finding the corresponding eigenvector by solving $(C-\lambda I)v = 0$. The eigenvectors will also be used to transform the original data into a reduced-dimension space.

1.6 Features:

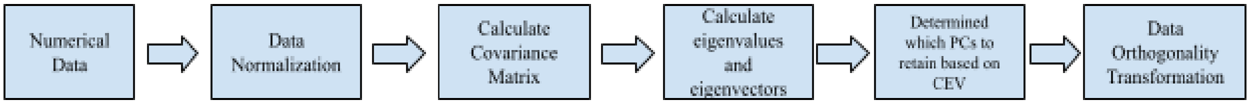
- PCA works well with high-dimensional datasets. It effectively helps to reduce the number of variables by transforming data to maintain only significant variables.
- It works well with the numerical data since its process is related with variance-covariance matrix computation.
- It effectively identifies the most significant features when data is standardized (with mean = 0 and variance = 1).

1.7 Guide: Inputs and Outputs

- Inputs : Numerical data with multiple features that need to be standardized to ensure that all features contribute equally. This prepared data will be used to compute the covariance matrix to determine the relationships between features.
- Outputs: Principal components (uncorrelated new variables), explained variance (the proportion of the variance retained by each component), transformed dataset with reduced dimensions.

1.8 Hyperparameters: PCA has no hyperparameter to find-tune during its process. However, to get the optimal components that capture the acceptance level of variance, a scree plot can be used to aid in this decision making.

1.9 Illustration: The flowchart below demonstrates the process for PCA implementation.


Numerical Data
Data Normalization
Calculate Covariance Matrix
Calculate eigenvalues and eigenvectors
Determined which PCs to retain based on CEV
Data Orthogonality Transformation

1.10 Journal: (Mbona and Yushen 233) implemented PCA to reduce the dimension of the financial ratios from 18 features to remain only 12 principal components that have 99.893% of the cumulative explained variance. The use of 12 new principal components extracted from the original features could reduce the complexity of the financial statement analysis while limiting the loss of information (243).

1.11. Keywords : Dimensionality reduction, Orthogonal transformation, Eigenvalues and Eigenvectors, Covariance Matrix .

2. K-Means Clustering

2.1 Advantages:

- Risk Assessment: One advantage of K-Means is its ability to group customers by risk levels, which drastically aids credit scoring and risk management to reduce risk.
- Asset Clustering: It helps to group similar assets through diversification and portfolio optimization to reduce financial loss.
- Performance Analysis: It is very useful to identify clusters of assets, especially high performing ones facilitating better decision-making.

2.2 Basics: K-Means Clustering is an Unsupervised Machine Learning algorithm and it works by grouping the unlabeled dataset into different clusters for effective performance. K-Mean is very crucial as it help to ensure the partitions of a dataset into K distinct clusters .K-means clustering is commonly used for: Clustering Algorithm development, Market Segmentation, Portfolio Risk Analysis, Trading Strategy development, Volatility Clustering, Stock Market Analysis

2.3 Computation: We used stock data for Apple (AAPL), Tesla (TSLA), Amazon (AMZN), Microsoft (MSFT), and Google (GOOG) from Yahoo Finance, covering the period from January 1, 2020, to January 1, 2024. We downloaded the 'Adjusted Close' prices using the Yahoo Finance API. From this data, we calculated daily returns and volatility, removed any NaN values, and calculated the standard deviation to measure risk. After normalizing the data, we applied k-means clustering.

2.4 Disadvantages:

- Difficulty Handling Non-Linear Structures : K-means clustering is poorly suited for handling non-linear structures, which are common in financial data. Its reliance on the assumption that clusters are convex makes it difficult to accurately analyze such complex datasets.
- Scalability Issues: K-means is often considered efficient for smaller datasets. Its performance degrades significantly with very large datasets. This is very common especially when the number of clusters (K) is high. Thereby resulting in increased computational costs.
- Unequal Cluster Sizes: K-means often assigns more points to larger clusters which leads to imbalanced clustering and reduced performance. This is a problematic in finance due to varying cluster densities
- Hard Assignments : The strict assignment of each data point to a single cluster in K-means makes it difficult to handle overlapping data distributions. For these situations, a Gaussian Mixture Model is a more appropriate choice.

2.5 Equations: K-means is a crucial clustering algorithm, simply because it helps to partition a dataset into K clusters by minimizing the sum of squared distances between data points and their assigned centroids, ensuring performance. The code is analyzed below.

$$J = \Sigma_{k=1}^{K} \Sigma_{x_i \in S_k} ||x_i - C_k||^2$$

where J represents the clustering cost (total cost)

K represents the number of clusters

X_i represents the data points assigned to each cluster K

C_k represents the centroid (centroid cluster K)

S_k represents the main set (set points in cluster K).

2.6 Features:

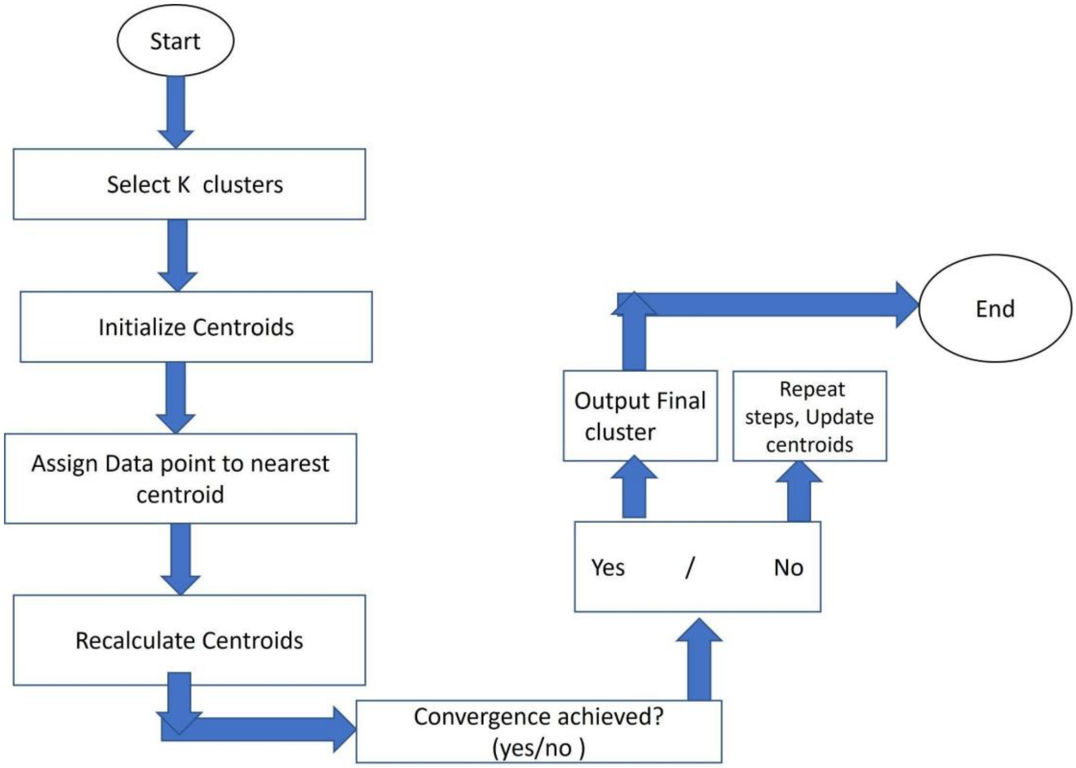
- It works well for large datasets, providing better performance than small datasets, which result in poor performance.
- It is very suitable and efficient for handling market segmentation and stock analysis, providing meaningful and accurate performance.
- It helps to normalize data for excellent performance, avoiding bias toward large numerical values.

2.7 Guide: List of inputs and outputs

- Inputs: A dataset with multiple numerical features requires clustering. To ensure this, it is important we define a predetermined number of clusters in order to group the data into. We also need to standardize the data to ensure all features contribute equally to clustering. Which is typically, euclidean distance is used to measure the potential similarity between points.
- Outputs: Each data point is assigned to a specific cluster. Then, the final cluster centroids are the center points of each cluster after it's done. The Within-Cluster Sum of Squares shows how tight the clusters are. Inertia is the total of squared distances from each data point to its cluster's center.

2.8 Hyperparameters: Key hyperparameters: It is crucial to understand that k, init, and n_init have the most significant effect on performance. These parameters ensure that the algorithm works accurately and effectively. Good initialization (k-means++) ensures better clusters and makes the clustering process perform better and faster. Fine-tuning max_iter and tol improves stability. It makes the algorithm work well. Silhouette Score helps to determine the best k.

2.9 Illustration: The flowchart below demonstrates the process for K-means clustering


Start
Select K clusters
Initialize Centroids
Assign Data point to nearest centroid
Recalculate Centroids
Convergence achieved? (yes/no)
Yes / No
Output Final cluster
Repeat steps, Update centroids
End

2.10 Journal: Fang & Chiao (2021) demonstrated that clustering techniques, specifically an optimized K-Means model, effectively categorize stocks into high-performance and low-performance groups. Their findings support the use of clustering for investment decision-making, and this also provides a foundation for further applications in portfolio optimization.

2.11 Keywords: Centroid-based Clustering, Partitioning Algorithm, Spherical Clustering , Iterative Algorithm , Unsupervised Learning , Euclidean Distance, K-Means++ Initialization , WCSS (Within-Cluster Sum of Squares)

3. LASSO regression

3.1 Advantages:

- Feature Selection: Automatically selects the most relevant variables by shrinking irrelevant ones to zero.
- Handles Multicollinearity: Reduces overfitting by removing redundant or highly correlated predictors.
- Improved Predictive Performance: Enhances generalization on new data by preventing overfitting.
- Sparse Solutions: Leads to more interpretable models by eliminating unnecessary variables.

3.2 Basics: LASSO (Least Absolute Shrinkage and Selection Operator) regression is a linear regression that tends to eliminate the weights of the least important features, setting them to zero. It achieves this by applying an L1 regularization penalty, which forces some coefficients to be exactly zero, effectively performing feature selection. LASSO is particularly useful in marketing analytics for identifying key predictors among numerous potential variables.

3.3 Computation: A comprehensive Python implementation using sklearn with dataset preprocessing, model fitting, hyperparameter tuning, and evaluation based on historical market data from SPY.

3.4 Disadvantages:

- Feature Selection: Automatically selects the most relevant variables by shrinking irrelevant ones to zero.
- Handles Multicollinearity: Reduces overfitting by removing redundant or highly correlated predictors.
- Improved Predictive Performance: Enhances generalization on new data by preventing overfitting.
- Sparse Solutions: Leads to more interpretable models by eliminating unnecessary variables.

3.5 Equations: LASSO minimizes the following objective function:

$$min_{\beta} \Sigma_{i=1}^{n} = (y_{i} - X_{i}\beta)^{2} + \lambda\Sigma_{i=1}^{p}|\beta_{j}|$$

where y is the dependent variable

X represents independent variables

β denotes regression coefficients

λ controls the regularization strength

3.6 Features:

- Customer Segmentation: Identifies key factors influencing customer behavior.

- Ad Spend Optimization: Determines the most impactful advertising channels.
- Price Sensitivity Analysis: Highlights how different pricing strategies affect sales.
- Churn Prediction: Helps identify factors contributing to customer attrition

3.7 Guide: Inputs and outputs:

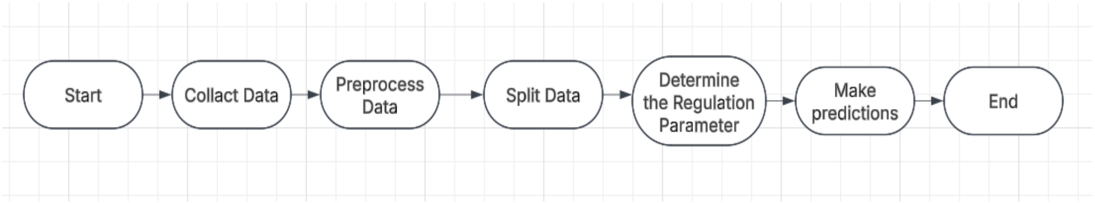
- Inputs: Predictor variables (X), Response variable (y), Regularization parameter (λ)
- Outputs: Estimated coefficients ($\widehat{\beta}$), Predicted variables ($\widehat{y}$)

In this document, the LASSO regression model takes historical market data from SPY, including Open, High, Low, Close, Volume, and Close prices. The dataset undergoes preprocessing, including standardization, and is split into training and testing sets. The model undergoes hyperparameter tuning using cross-validation to determine the best regularization strength (λ).The trained LASSO model provides feature selection by shrinking some coefficients to zero, generates predictions for the test dataset, and evaluates performance using Mean Squared Error (MSE) and R-Squared (R^2) Score.

3.8 Hyperparameters

- Choosing the Optimal λ: Performed using cross-validation techniques like Grid Search or Randomized Search.
- Trade-off Consideration: Higher λ increases sparsity but may reduce accuracy.
- LASSO Path Analysis: Examining how coefficients change with varying λ values.

3.9 illustration:



3.10 Journal:

This foundational paper introduces LASSO regression, discussing its feature selection ability and how it handles multicollinearity.

- Tibshirani, Robert. “Regression Shrinkage and Selection via the Lasso.” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, 1996, pp. 267–288. JSTOR.

This resource explains how LASSO reduces overfitting and enhances model interpretability.

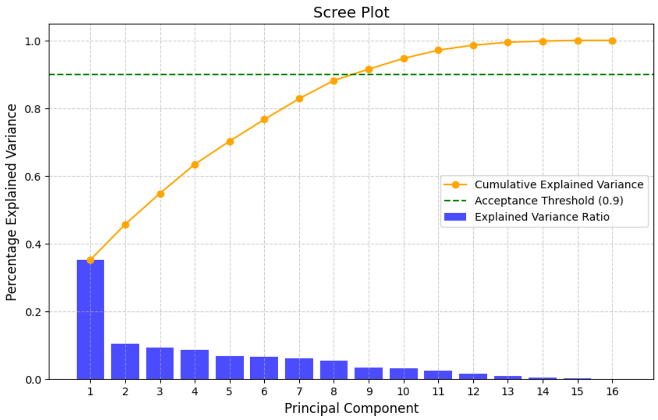
- Hastie, Trevor, et al. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* 2nd ed., Springer, 2009.

3.11 keyworks: Sparse regression, feature selection, marketing analytics, predictive modeling, L1 regularization.

Task 2 Technical Section

1. Principal Component Analysis

Based on the principal component analysis's algorithm, there is no hyperparameter that we needed to find-tune when we implemented this algorithm. However, determining the appropriate number of principal components remains a challenge when we interpret the results.

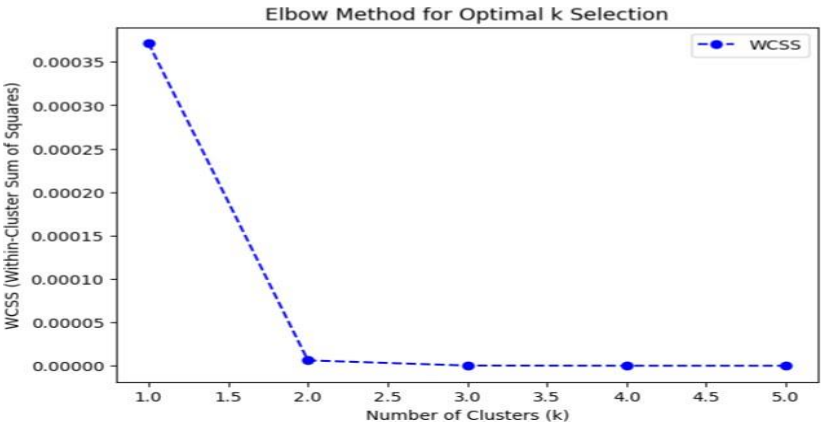


To identify the appropriate number of principal components we may set the acceptance threshold and calculate the cumulative explained variance. The appropriate number of principal components is determined when the cumulative explained variance is greater than or equal to the acceptance threshold.

From the data we used in section 1.1, we plotted the explained variance and accumulated explained variance as demonstrated in the scree plot below. With the acceptance threshold set at 0.9, the appropriate number of principal components for this dataset is 9 components which have cumulative explained variance at 91.52%.

2. K-means Clustering

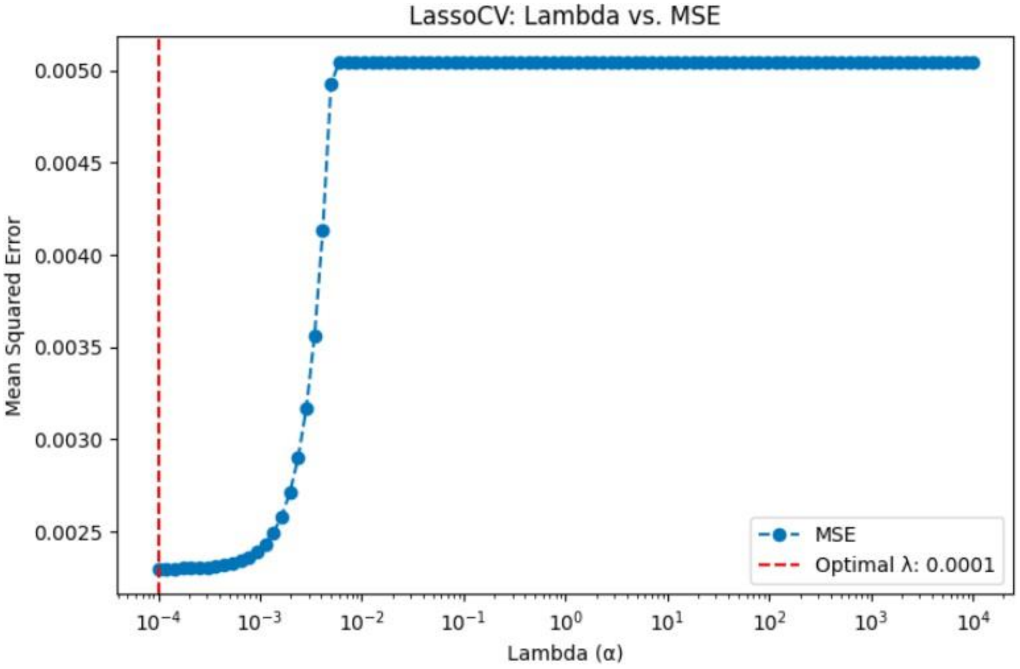
In the k-means clustering algorithm, there are many hyperparameters that can drastically influence the clustering quality. One of the main challenges we experienced was determining the right number of clusters (K) that works accurately for the dataset to perform well. In order to identify the right number of clusters for the Model to perform well, we simply use the Elbow Method, which gives us the right way to analyze the within-cluster sum of squares (WCSS). It also helps us to find where adding more clusters does not give an issue that is reducing the Within-Cluster Sum of Squares (WCSS). The optimal k is determined at the point where the WCSS curve forms an "elbow."



From the dataset used in Section 2.3, we plotted the WCSS against different values of k, as demonstrated in the Elbow Method plot. Based on this plot, we observe that the optimal number of clusters is 2, where the curve starts to flatten, indicating minimal improvement beyond this point.

3. LASSO regression

Regarding LASSO regression is a type of linear regression, when tuning it, key hyperparameters must be carefully optimized to achieve the best model performance. Using Grid Search and Randomized Search for hyperparameter tuning. It also includes cross-validation and visualization of the tuning process. In this process, applying LASSO regression to spy stock price data to predict returns using historical feature, while optimizing the regularization parameter (λ) through cross-validation and hyperparameter tuning. LassoCV automatically selects the best λ from a logarithmic range (10^{-4} to 10^{4}) using 5-fold cross-validation.The best λ is identified by minimizing Mean Squared Error (MSE).

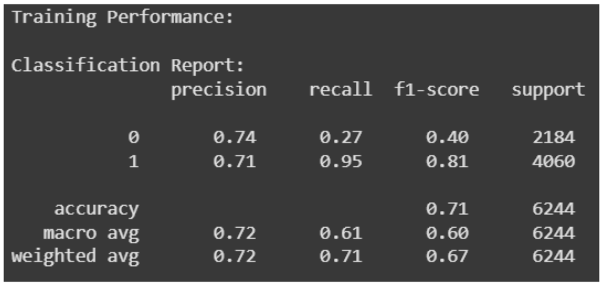
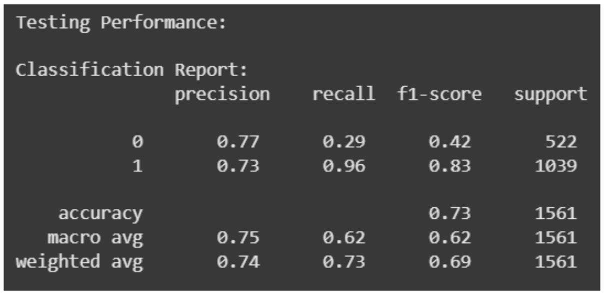


Task 3 Marketing Alpha

1. Principal Component analysis (PCA)

The main advantage of the PCA is its ability to reduce high -dimensional features into smaller essential features. This helps to generalize the model, improve model predictive performance, and lower the computational cost of the model's parameters' estimation.

Based on the data of corporate credit rating with financial ratios available at Kaggle, we want to use 18 financial ratios to train the binary classification model for the investment grade bond. We implemented PCA to reduce 18 original features into 9 principal components that have a cumulative explained variance equal to 91.97%. Then, we train the logistic regression with the transformed features. The model performance metrics are similar for both training and testing data. This indicates that the estimated model is well generalized. Hence, PCA could help to avoid overfitting by focusing on selecting the most important features and maintaining consistent performance across training and testing data.

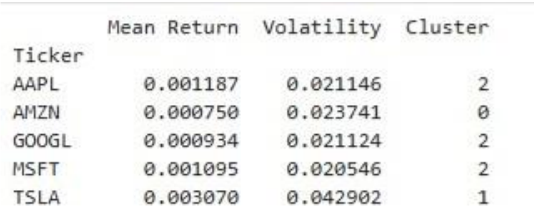
Training Data	Testing Data
	

```
Training Performance:

Classification Report:
              precision    recall  f1-score   support

           0       0.74      0.27      0.40      2184
           1       0.71      0.95      0.81      4060

    accuracy                           0.71      6244
   macro avg       0.72      0.61      0.60      6244
weighted avg       0.72      0.71      0.67      6244
```

```
Testing Performance:

Classification Report:
              precision    recall  f1-score   support

           0       0.77      0.29      0.42       522
           1       0.73      0.96      0.83      1039

    accuracy                           0.73      1561
   macro avg       0.75      0.62      0.62      1561
weighted avg       0.74      0.73      0.69      1561
```

2. K-mean Clustering

One of the main advantages of K-Means is its ability to classify similar stocks based on their mean return and volatility. This can simply help investors to make prompt decisions by distinguishing between high risk, high reward stocks and stable investment. Based on the financial data downloaded from Yahoo Finance. We use the closing prices from AAPL, TSLA, AMZN, MSFT, and GOOGL from 2020 to 2024. We can train and test K-Means in order to uncover patterns in stock performance. K-Means clustering is seen as a useful technique for analyzing financial data, especially when we are using it to identify patterns in stock market trends based on factors such as volatility and returns. Our analysis demonstrates that by grouping stocks into clusters. This can help investors to tailor their portfolios in order to align with their specific risk tolerance. After running a K-means clustering each stock is assigned a cluster based on its similarity to others as shown in the table below.

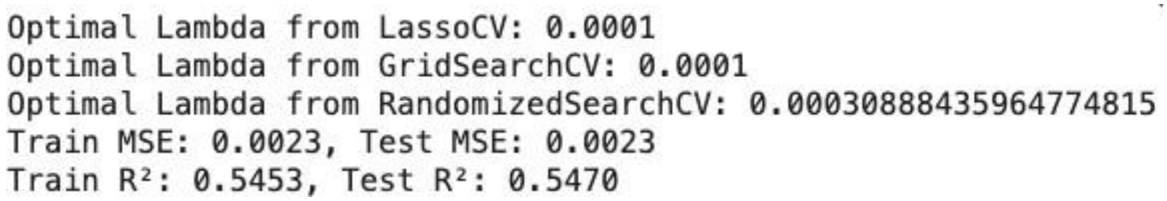


```
        Mean Return  Volatility  Cluster
Ticker
AAPL       0.001187    0.021146        2
AMZN       0.000750    0.023741        0
GOOGL      0.000934    0.021124        2
MSFT       0.001095    0.020546        2
TSLA       0.003070    0.042902        1
```

High-volatility clusters may contain stocks like APPL(2) , GOOGL(2) and MSFT(2) due to large price fluctuations. Stable clusters may include TSLA(1) which typically show steady growth. Growth-focused clusters may feature stocks like AMZN(0), which historically has long-term upward trends.

3. LASSO regression

In ML-enhanced LASSO regression, Using historical market data to predict SPY ETF returns, which include open, high, low, close, and volume data. Open and Close prices contribute the most to return predictions. Model performance is evaluated using Mean Squared Error (MSE) and R-Squared (R²) Score. The best-performing LASSO model achieves an MSE of 0.0023 and an R² score of 0.5470, indicating a predictive ability.



```
Optimal Lambda from LassoCV: 0.0001
Optimal Lambda from GridSearchCV: 0.0001
Optimal Lambda from RandomizedSearchCV: 0.00030888435964774815
Train MSE: 0.0023, Test MSE: 0.0023
Train R²: 0.5453, Test R²: 0.5470
```

Task 4 Learn More

1. Principal Component Analysis

The main advantage of the principal components is to reduce the large numerical features in the dataset into the essential smaller features called principal components. We found that the following studies demonstrated the roles of PCA not only reducing the high dimension features but also enhancing the models' performance.

(Mbona and Yusheng 243) applied the PCA to perform the financial statement analysis for China telecoms industry. Since some financial ratios have similar information, the PCA was implemented to reduce 18 financial ratios into only 12 principal components. This helps to reduce the complexity of the firms' performance evaluation based on the financial ratios.

(Yu, Chen, and Zhang 411) conducted the stock selection by implementing the support vector machine algorithm (SVM). The PCA was used to reduce the high dimension features of the financial ratios to be a smaller number of features. The results of implementing PCA and SVM could enhance the model accuracy and efficiency in the stock selection task. The constructed portfolio based on this PCA-SVM algorithm could outperform the A-share index of Shanghai Stock Exchange that had been used as a benchmark.

(Nobre and Neves 181) proposed the stock picking algorithm by combining PCA with other machine learning algorithms. First, the PCA was implemented to reduce the financial input dataset, then the discrete wavelet transformation was used to perform noise reduction for the highly contributed features from PCA. This processed input data was implemented with XG-Boost to train the binary classification model for the stock selection. The implementation of PCA greatly enhanced the model performance (189).

(Chowdhury, Rayhan, Chakravary and Hossain 28) found that the PCA could be efficiently implemented with the time-series stock price forecasting. The 16 technical indicators inputs were reduced to be 10 components with PCA which contribute over 98% of the cumulative variance of the data (32). Then these components were used as the input for the support vector regression to forecast the future stock price. The PCA-SCR model could outperform simple SVR in the forecasting measured by various forecasting performance metrics such as mean squared error, root mean squared error, and mean absolute error (31).

(Mavungu 1) developed the financial index for measuring the financial health of the listed companies in the Alternative exchange of Romania. Eight financial ratios were selected as the independent variables. The PCA was implemented to reduce these independent variables into three main components. The selected financial ration had been transformed by the eigenvectors and constructed new features called return rates, liquidity rate, and management rates (9). Then the panel regression was implemented with these three independent variables to study the relationship with solvency rate. The results demonstrated that these three principal components had a significant positive relationship with solvency rate and PCA could help to reduce the model complexity (12).

2. K-mean clustering

One of the main advantages of K-Means is its ability to group similar data points based on their mean return and volatility. This can simply help investors to make prompt decisions by distinguishing between high risk, high reward stocks and stable investment. Furthermore, previous studies have shown that K-Means not only groups similar data points but also serves as a method to minimize variance within clusters, helping investors make decisions.

(Wedel and Kamakura) provides a broader overview of segmentation methodologies, including traditional clustering algorithms like K-Means and advancements in finite mixture models. The study specifically highlights the importance and advantages of K-Means Clustering in segmenting customers based on specific behaviors. It helps us to identify customer segments based on investment behaviors and risk preferences. The K-means clustering provides a method for determining how to minimize investment risk for each customer segment. Following this approach provides a method for minimizing investment risk.

Researchers report that "the k-means clustering algorithm was used to cluster the stocks and divide the different types of stock pools" (Wu, Xiaolong , and Shaocong). It is a popular machine learning method used to group similar stocks based on their features. Thereby, helping investors to identify which stocks are more or less risky. It provides a method of identifying patterns among stocks. K-Means clustering is a powerful tool because it analyzes various stocks and groups them into clusters based on their performance and trends. This therefore reduces financial loss for investors and allows them to focus on lower risk stocks. By clustering the stocks, the algorithm creates distinct categories of stocks. Each category may consist of stocks that behave similarly in the market. This would enable investors to analyze and select stocks for their portfolios.

(Bhattacharyya, Sanjeev, Kurian and Westland) evaluate advanced data mining approaches for credit card fraud detection, including support vector machines, random forests, and logistic regression while K-Means offers an advanced alternative. This study presents K-Means as an effective technique for identifying fraudulent transactions by clustering abnormal spending patterns in real-time financial data. Any fraudulent transaction can be easily detected with the help of this algorithm, making it easy to prevent any form of abnormal spending patterns in real-time financial data. This enables the limitation of financial losses, ensuring that investors would no longer worry about losing their funds. Like other credit card fraud detection methods, K-Means offers an advantageous alternative for clustering abnormal spending and detecting fraudulent transactions within an organization.

3. LASSO regression

The advantage of LASSO regression is that it can efficiently identify the most relevant variables, and reduce the irrelevant to zero. In this process, LASSO regression is reducing overfitting by removing highly correlated predictors that could otherwise destabilize the model. So,with proven predictive performance, LASSO can ensure the model generalizes better to unseen data, avoiding overly complex fits to the training dataset. Moreover, LASSO greatly enhances interpretability with its sparsity of the solutions , allowing researchers and practitioners to clearly identify influential predictors and thus derive meaningful insights from simpler, more transparent models.

(Tibshirani)'s foundational work introduces the LASSO regression method, which demonstrates how effectively it combines regularization and feature selection through an L1-penalty. LASSO shrinks certain regression coefficients precisely to zero by imposing this penalty, as well as automatically eliminating irrelevant or redundant predictors. This feature can both result in select capability and addresses multicollinearity by managing correlated predictors, reducing instability, and enhancing overall model robustness. By theoretical derivations and empirical simulations, Tibshirani validates that LASSO provides more stability and interpretability than traditional regression methods, particularly advantageous for high-dimensional datasets with numerous variables.

(Hastie, Tibshirani, and Friedman) illustrate the strengths of LASSO regression, by emphasizing its capacity to reduce overfitting through coefficient shrinkage, thereby enhancing the model's ability to generalize effectively to new data.Their textbook thoroughly highlights how LASSO-generated sparse solutions significantly enhance interpretability, empowering analysts to precisely identify and focus on the most impactful predictors. The authors demonstrate LASSO's exceptional effectiveness, especially within high-dimensional scenarios involving multiple correlated or irrelevant predictors. Utilizing real-world examples discussed in the text further underscore LASSO's effectiveness in producing accurate predictions alongside simpler, highly interpretable models, reinforcing its practical value in statistical learning and informed, data-driven decision-making.

References

Bhattacharyya, Siddhartha, Sanjeev Jha, Kurian Tharakunnel, and Westland, J. Christopher. "Data mining for credit card fraud: A comparative study." *Decision support systems* 50.3 (2011): 602-613.

Cui, Mengyao. "Introduction to the k-means clustering algorithm based on the elbow method." *Accounting, Auditing and Finance* 1.1 (2020): 5-8.

Chowdhury, Utpala Nanda, et al. "Integration of principal component analysis and support vector regression for financial time series forecasting." *International Journal of Computer Science and Information Security (IJCSIS)* 15.8 (2017): 28-32.

Fang, Zheng, and Chaoshin Chiao. "Research on prediction and recommendation of financial stocks based on K-means clustering algorithm optimization." *Journal of Computational Methods in Science and Engineering* 21.5 (2021): 1081-1089.

Gwerc, Alexandre. *Corporate Credit Rating Forecasting*. Kaggle, https://www.kaggle.com/code/agewerc/corporate-credit-rating-forecasting . Accessed 15 Mar. 2025.

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. *The elements of statistical learning: data mining, inference, and prediction*. 2nd., New York: Springer,2009.

Mavungu, Masiala. "Computation of financial risk using principal component analysis." *Algorithmic Finance* 10.1-2 (2023): 1-20.

Mbona, Reginald Masimba, and Kong Yusheng. "Financial statement analysis: Principal component analysis (PCA) approach case study on China telecoms industry." *Asian Journal of Accounting Research* 4.2 (2019): 233-245.

Nobre, João, and Rui Ferreira Neves. "Combining principal component analysis, discrete wavelet transform and XGBoost to trade in the financial markets." *Expert Systems with Applications* 125 (2019): 181-194.

Tibshirani, Robert. “Regression Shrinkage and Selection via the Lasso.” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, 1996, pp. 267–288. JSTOR.

Wedel, Michel, and Wagner A. Kamakura. *Market segmentation: Conceptual and methodological foundations*. Springer Science & Business Media, 2000.

Wu, Dingming, Xiaolong Wang, and Shaocong Wu. "Construction of stock portfolios based on k-means clustering of continuous trend features." *Knowledge-Based Systems* 252 (2022): 109358.

Yu, Huanhuan, Rongda Chen, and Guoping Zhang. "A SVM stock selection model within PCA." *Procedia computer science* 31 (2014): 406-412.